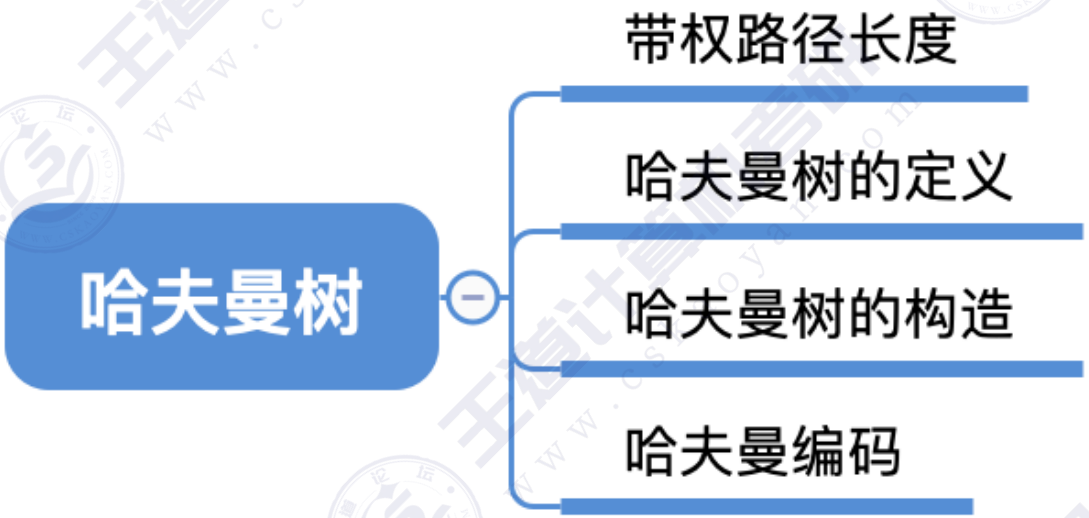
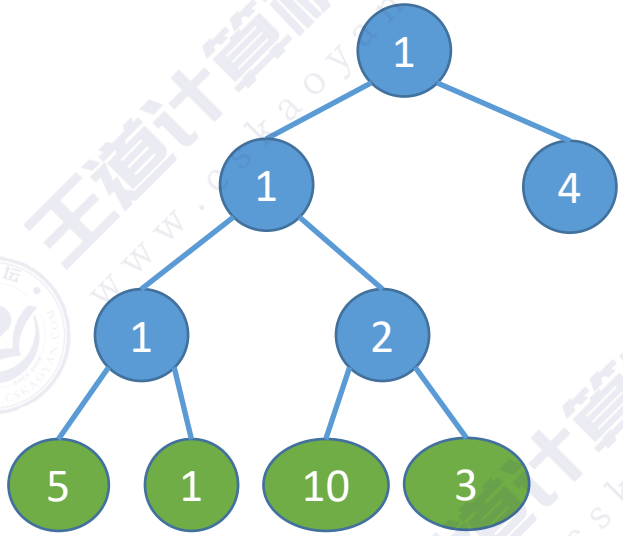
本节内容

哈夫曼树

知识总览



带权路径长度



结点的权：有某种现实含义的数值（如：表示结点的重要性等）

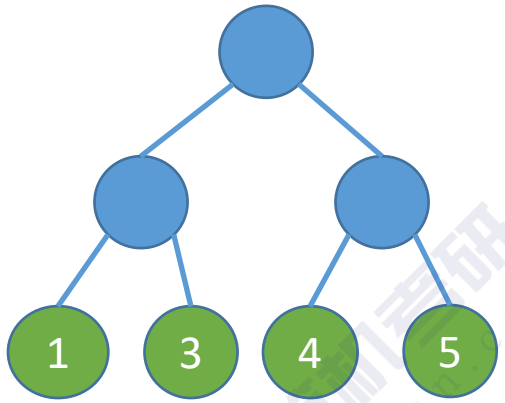
结点的带权路径长度：从树的根到该结点的路径长度（经过的边数）与该结点上权值的乘积

树的带权路径长度：树中所有叶结点的带权路径长度之和（WPL, Weighted Path Length）

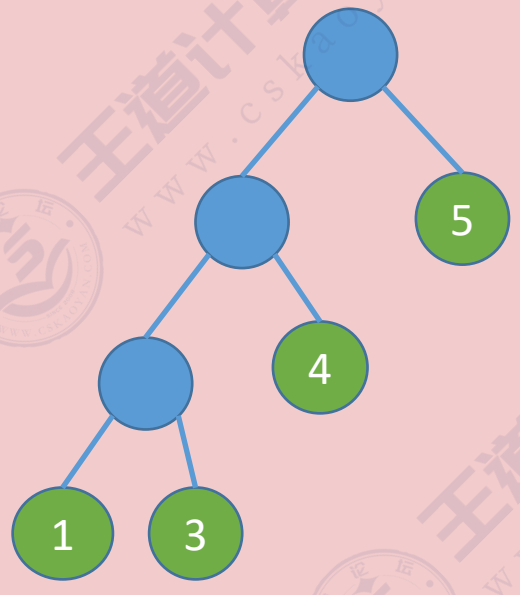
$$\mathrm{WPL}=\sum_{i=1}^{n} w_i l_i$$

哈夫曼树的定义

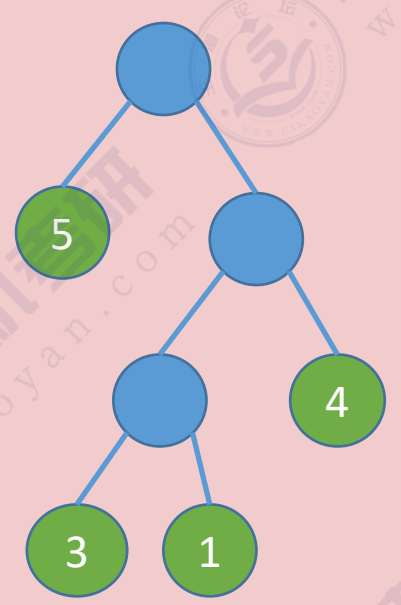
都是哈夫曼树



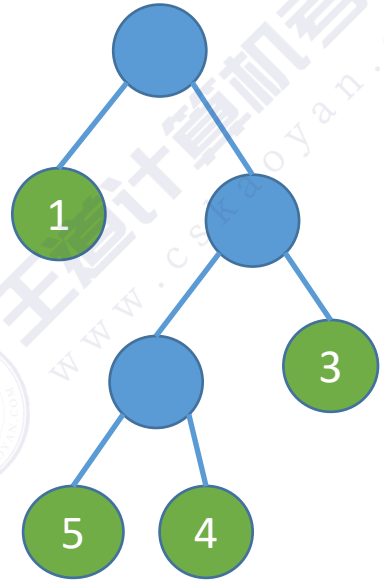
WPL= 2*1+2*3+2*4+2*5 = 26



WPL=1*5+2*4+3*1+3*3=25



WPL=1*5+2*4+3*1+3*3=25



WPL=1*1+2*3+3*5+3*4=34

在含有*n*个带权叶结点的二叉树中，其中带权路径长度（WPL）最小的二叉树称为哈夫曼树，也称最优二叉树

哈夫曼树的构造

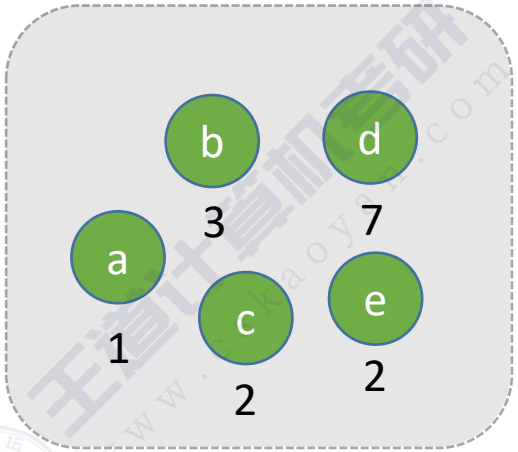
给定n个权值分别为$w_1, w_2, \ldots, w_n$的结点，构造哈夫曼树的算法描述如下：

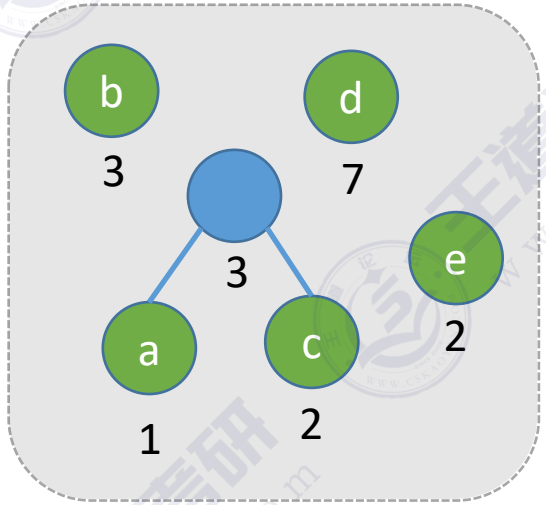
1）将这n个结点分别作为n棵仅含一个结点的二叉树，构成森林F。

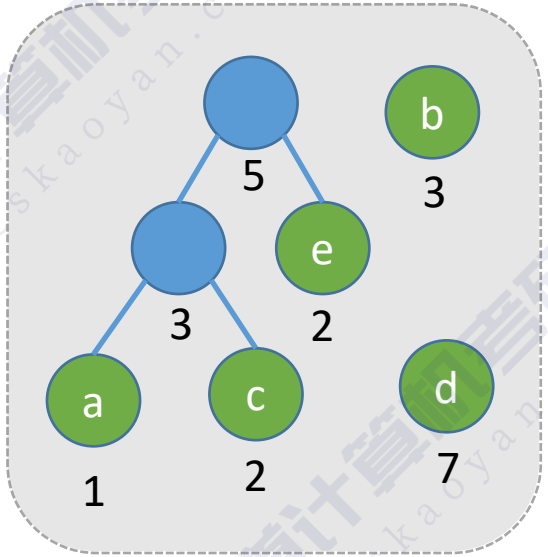
2）构造一个新结点，从F中选取两棵根结点权值最小的树作为新结点的左、右子树，并且将新结点的权值置为左、右子树上根结点的权值之和。

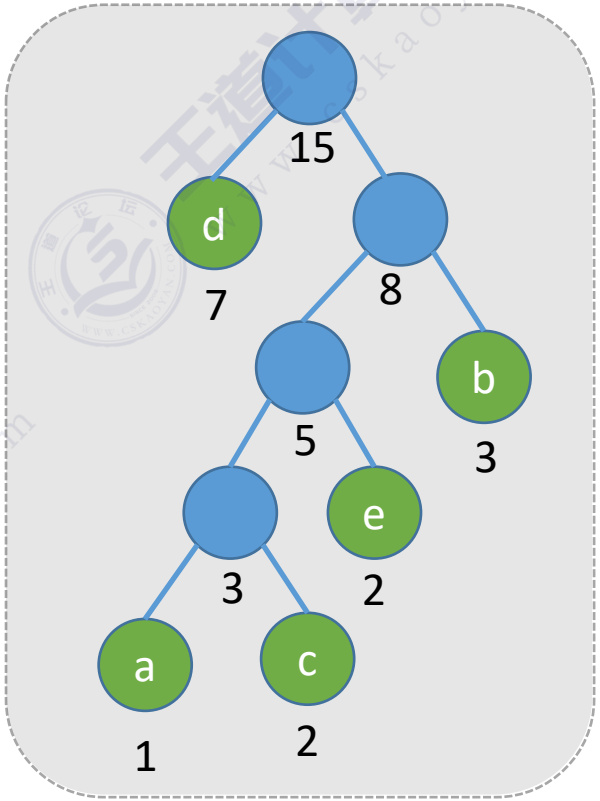
3）从F中删除刚才选出的两棵树，同时将新得到的树加入F中。

4）重复步骤2）和3），直至F中只剩下一棵树为止。









1）每个初始结点最终都成为叶结点，且权值越小的结点到根结点的路径长度越大

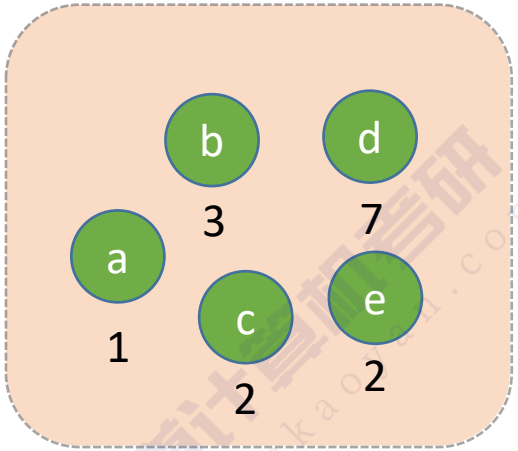
2）哈夫曼树的结点总数为$2n-1$

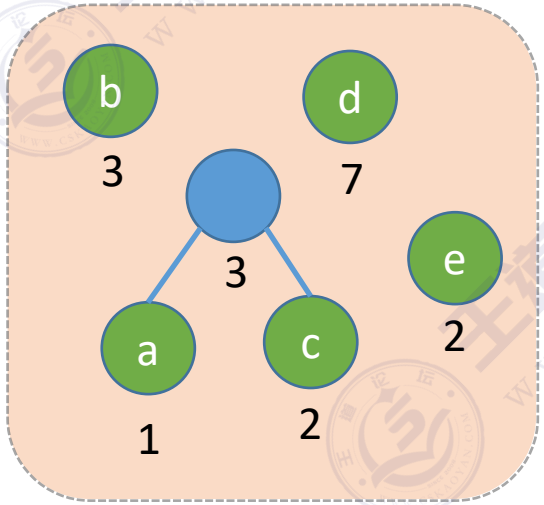
3）哈夫曼树中不存在度为1的结点。

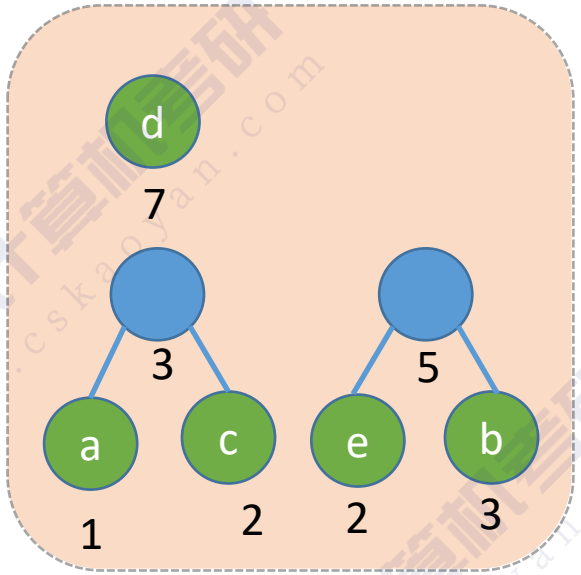
4）哈夫曼树并不唯一，但WPL必然相同且为最优

$WPL_{min}=1*7+2*3+3*2+4*1+4*2=31$

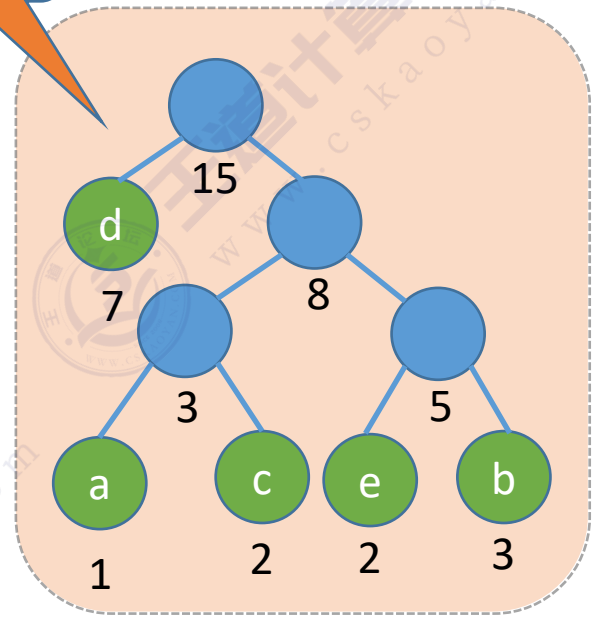
哈夫曼树的构造







左右顺序任意



WPL=1*7+3*(1+2+2+3)=31

哈夫曼编码

电报——点、划 两个信号（二进制0/1）

哈夫曼编码

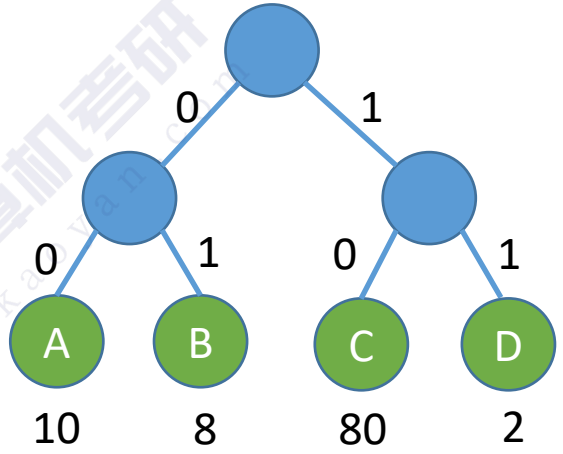
固定长度编码——每个字符用相等长度的二进制位表示

A——0100 0001
B——0100 0010
C——0100 0011
D——0100 0100

ASCII编码

A——00
B——01
C——10
D——11

每个字符用长度为2的二进制表示



假设，100题中有80题选C，10题选A，8题选B，2题选D

所有答案的二进制长度=80*2+10*2+8*2+2*2=200 bit

WPL= 80*2+10*2+8*2+2*2=200

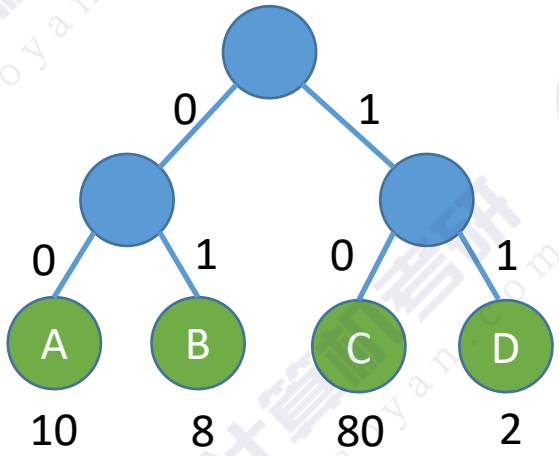
哈夫曼编码

固定长度编码——每个字符用相等长度的二进制位表示

A——00
B——01
C——10
D——11

每个字符用长度为2的二进制表示

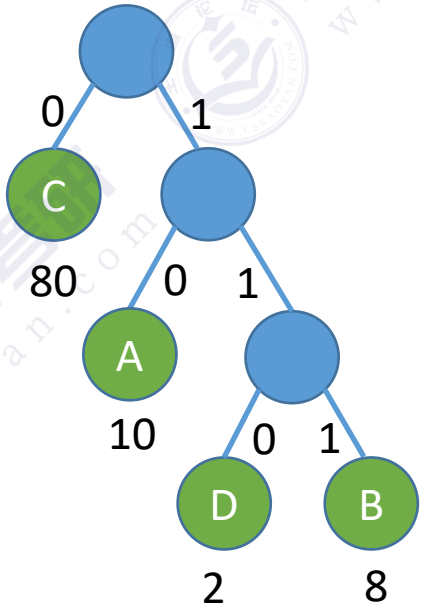
假设，100题中有80题选C，10题选A，8题选B，2题选D
所有答案的二进制长度=80*2+10*2+8*2+2*2=200 bit



WPL= 80*2+10*2+8*2+2*2=200

C——0
A——10
B——111
D——110

可变长度编码——允许对不同字符用不等长的二进制位表示

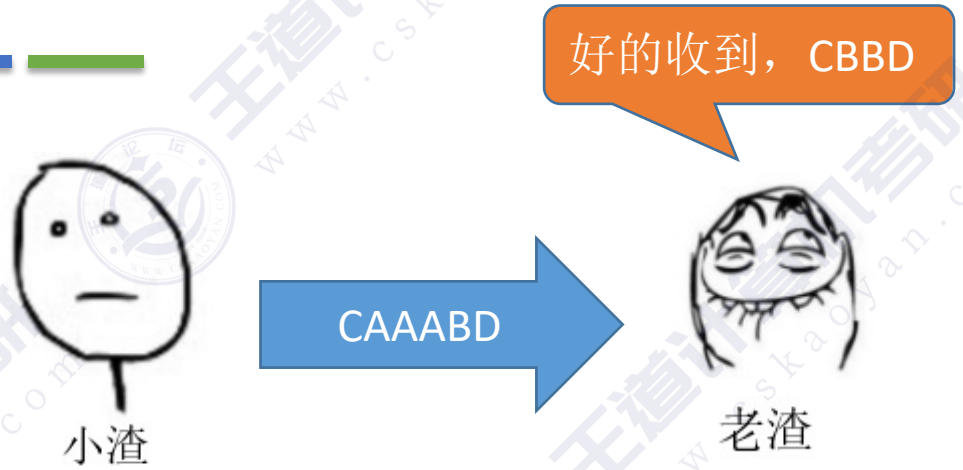


WPL= 80*1+10*2+2*3+8*3=130

哈夫曼编码

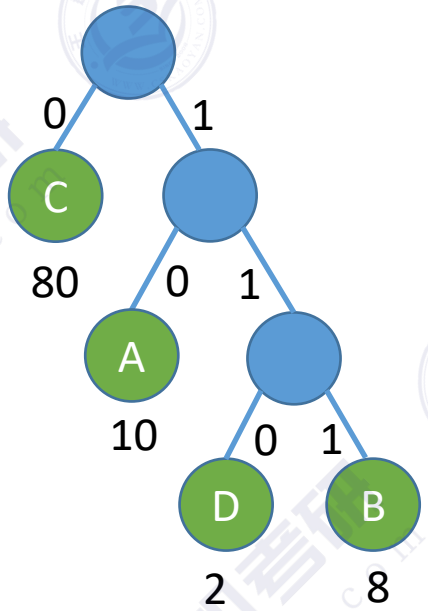
可变长度编码——允许对不同字符用不等长的二进制位表示

若没有一个编码是另一个编码的前缀，则称这样的编码为前缀编码



前缀码解码无歧义

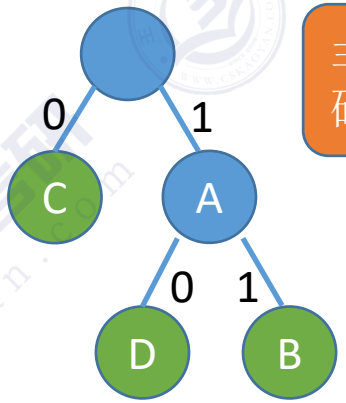
C——0
A——10
B——111
D——110



WPL= 80*1+10*2+2*3+8*3=130

CAAABD：0101010111110

C——0
A——1
B——111
D——110



非前缀码解码有歧义

CAAABD：0111111110

哈夫曼编码

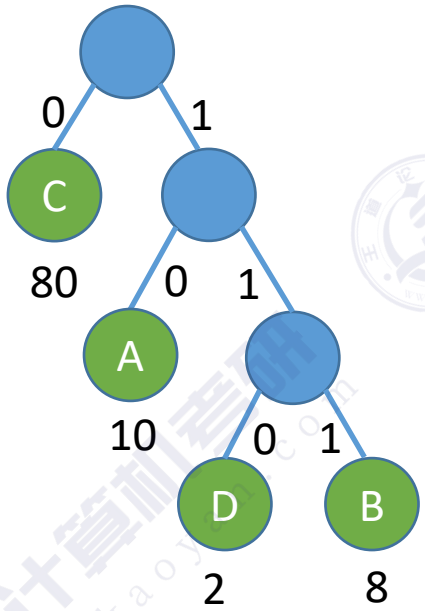
固定长度编码——每个字符用相等长度的二进制位表示

可变长度编码——允许对不同字符用不等长的二进制位表示

若没有一个编码是另一个编码的前缀，则称这样的编码为前缀编码

有哈夫曼树得到哈夫曼编码——字符集中的每个字符作为一个叶子结点，各个字符出现的频度作为结点的权值，根据之前介绍的方法构造哈夫曼树

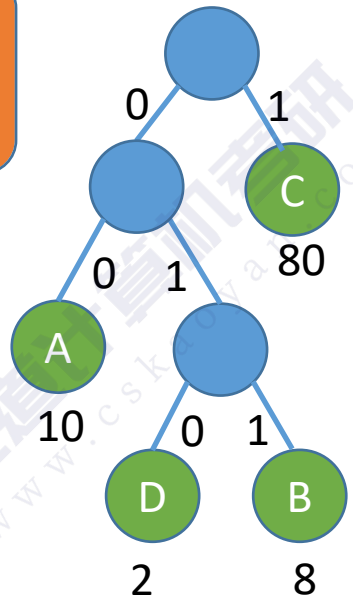
C——0
A——10
B——111
D——110



WPL= 80*1+10*2+2*3+8*3=130

哈夫曼树不唯一，因此哈夫曼编码不唯一

C——1
A——00
B——011
D——010



WPL= 80*1+10*2+2*3+8*3=130

哈夫曼编码可用于数据压缩

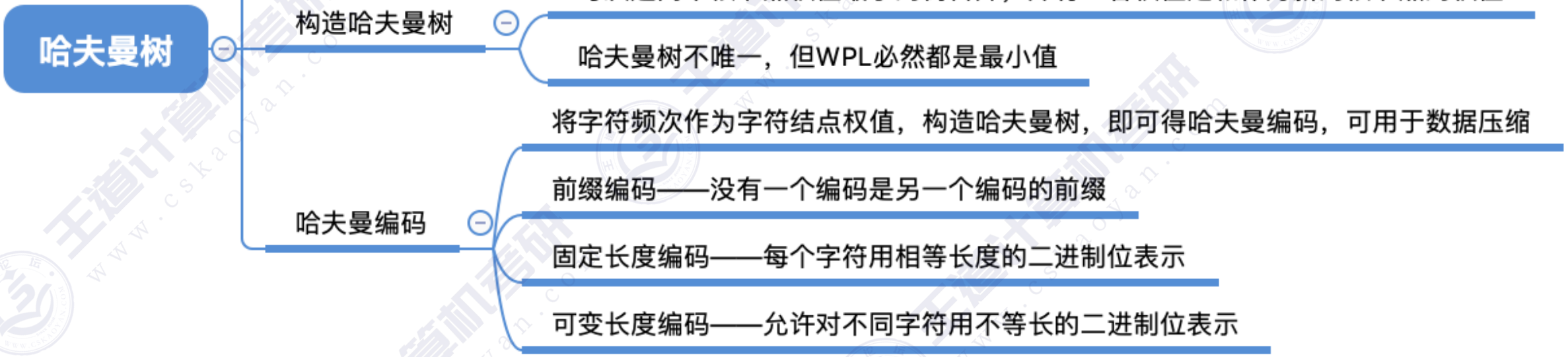
英文字母频次

英文字母使用频率表:(%)

A 8.19	B 1.47	C 3.83	D 3.91	E 12.25	F 2.26	G 1.71
H 4.57	I 7.10	J 0.14	K 0.41	L 3.77	M 3.34	N 7.06
O 7.26	P 2.89	Q 0.09	R 6.85	S 6.36	T 9.41	
U 2.58	V 1.09	W 1.59	X 0.21	Y 1.58	Z 0.08	

试试设计哈夫曼编码，并计算数据压缩率

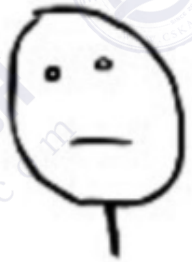
知识回顾与重要考点



结局

可变长度编码——允许对不同字符用不等长的二进制位表示

若没有一个编码是另一个编码的前缀，则称这样的编码为前缀编码

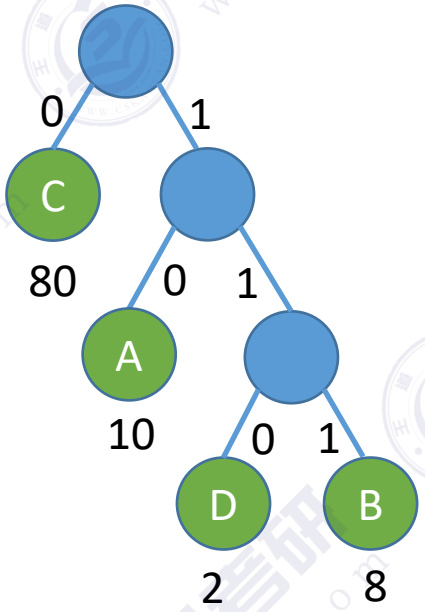

小渣

100道选择题


老渣

C——0
A——10
B——111
D——110

哈夫曼编码



WPL= 80*1+10*2+2*3+8*3=130 bit